



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/731,634	12/09/2003	Marcin Sawicki	60001.264US01	4750

27488 7590 06/18/2007
MERCHANT & GOULD (MICROSOFT)
P.O. BOX 2903
MINNEAPOLIS, MN 55402-0903

EXAMINER

TRUONG, LECHI

ART UNIT	PAPER NUMBER
----------	--------------

2194

MAIL DATE	DELIVERY MODE
-----------	---------------

06/18/2007

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

10/731,634

Applicant(s)

SAWICKI ET AL.

Examiner

LeChi Truong

Art Unit

2194

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 09 December 2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-41 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-41 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau. (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date 05/08/2006.
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-41 are presented for the examination.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

2. Claims 1-4 are rejected under 35 U.S.C. 101 because they are directed to non-statutory subject matter.

3. Claims 1-4 are non-statutory because they are software per se embodied in a manner so as to be executable.

Claims 1-4 define "A programmable object mode" in the preamble and the body of the claim recites "a application programming interface". An application programming interface appears to be software module. Therefore, claims 1-4 are non-statutory because they recite system claims that comprise software embodiments that do not have the use of hardware computer.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

Art Unit: 2194

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claims 1-4 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lee at al (US 2003/0145197 A1) in view of Nussbaum et al (US 6,779,154 B1).

As to claim 1, Lee teaches the invention substantially as claimed including: An application programming interface (the analyzing/processing unit 150 is called, para [0032], ln 1-3, a user (user, para [0032], ln 1-3), the schema validation system (the analyzing/processing unit 150 calls the illegitimate change detecting file generating unit 160, para [0032], ln 1-3 and ln 17-22), an application programming interface for allowing a user to programmatically access the functionality of the schema validation system(para[0032], ln 1-4 and ln 17-22); a message call(calls, para[0032], ln 9-16), an file(the illegitimate change detecting information, para[0032], ln 9-17), document(documents and resources, para[0008], ln 28-32/ ln 39-43),the application programming interface comprising a message call for requesting association of an XML schema file to a document(para[0032], ln 9-17), a return value(displays the resources linked to corresponding web page to user, para[0032], ln 1-7), association of the XML schema file to the document(receives from the user interface 110 to which resource the illegitimate detecting information processing is performed and to which resource the encryption is applied, [0032, ln 5-10), and the application programming interface operative to receive a return value from the schema validation system responsive to association of the XML schema file to the document(para[0032], ln 1-10).

Art Unit: 2194

Lee does not teach file as a schema XML file. However, Nussbaum teaches a schema XML file (inserts the text of the selected XML document in its entirety into the new HTML page, col 7, ln 51-54).

It would have been obvious to one of ordinary skill in the art at the invention was made to modify the teaching of Lee to incorporate the feature of schema XML because this enables the content information and linking information for an XML document during execution to be preserved for analysis using web development tools.

As to **claims 2, 3, 4**, they are apparatus claims of claim 1; therefore, they are rejected for the same reasons as claim 1. In additional, Lee teaches receive return value associate with providing access/ providing a location/ applied to (para [0032], ln 1-15).

5. Claims **5-7** are rejected under 35 U.S.C. 102(e) as being anticipated by Chau et al (US 6,643,633 B2).

As to **claim 5**, Lee teaches the functionality of an Extensible Markup Language (XML)(XML system's stored procedures, col 8, ln 55-57/ the XML system implemented by stored procedures, col 36, ln 42-47), call (called, col 8, ln 55-57/ col 36, ln 42-47), an object-oriented message call (col 29,ln 40-42), calling the software application via an object-oriented message call (col 8, ln 55-57/col 36, ln 42-47), an object property (a data Access Definition (DAD), col 36, ln 42-48/ col 41, ln 35-40), passing an object property to the software application(col 36, 42-48/col 41, ln 35-40), the object property being associated with XML functionality of the software application(col 83, ln 40-45); in response to the message call and

Art Unit: 2194

the object property passed to the software application, receiving access to the XML functionality of the software application associated with the object property passed to the software application(col 41, ln 62-67).

As to **claim 6**, Chau teaches passing the object property to the software application includes passing an object method to the software application for transforming an XML document (col 8, ln 48-53), the method object including as a parameter a path to an XSLT transformation file for transforming the XML document according to the XSLT transformation file (col 9, ln 37-47/ col 7, ln 62-67).

As to **claim 7**, Chau teaches whereby the method object passed to the software application further includes a parameter indicating whether the XSLT transformation file is to be applied to all data contained in the XML file or whether the XSLT transformation file is to be applied to only non-native XML markup data applied to the XML document (col 54, ln 25-32col 56, ln 5-10).

6. Claims **8-41** are rejected under 35 U.S.C. 103(a) as being unpatentable over Chau et al (US 6,643,633 B2) in view of Nussbaum et al (US 6,779,154 B1)

As to **claim 8**, Chau teaches whereby passing the object property to the software application includes passing an object property to the software application for controlling the appearance of an associated text when an XML element associated with text is to be presented in the software application's user Interface (col 24, ln 45-50/ col 8, ln 32-35).

Chau does not explicitly teach XML file. However, Nussbaum teaches a schema XML file (inserts the text of the selected XML document in its entirety into the new HTML page, col 7, ln 51-54).

It would have been obvious to one of ordinary skill in the art at the invention was made to modify the teaching of Chau to incorporate the feature of schema XML because this enables the content information and linking information for an XML document during execution to be preserved for analysis using web development tools.

As to claim 9, Chau teaches passing the object property to the software application includes passing an object property for controlling how the software application saves XML markup applied to a document (col 7, ln 53-61).

As to claim 10, Chau teaches passing the object property for controlling how the software application saves the XML markup applied to the document includes passing an object property for causing the software application to save the XML markup applied to the document according to the native XML functionality of the software application without application of any XSLT transformation prior to saving the XML markup applied to the document (col 7, ln 53-61).

As to claim 11, Chau teaches passing the object property to the software application includes passing an object property to the software application for causing the application to save only XML markup and associated data applied to the document (col 8, ln 46-57).

As to claim 12, Chau teaches passing the object property to the software application includes passing an object property to the software application for causing the software application to apply and XSLT transformation to the XML markup and associated data applied to

Art Unit: 2194

the document and any non-XML data contained in the document prior to saving the document (col 7, ln 53-61).

As to claim 13, Chau teaches passing the object property to the software application includes passing the object property for causing the software application to apply the XSLT transformation to only the non-XML data contained in the document prior to saving the XML document (col 7, ln 53-61).

As to claim 14, Chau teaches passing an object property to the software application includes passing an object property for controlling the location of an XSLT transformation to be applied by the software application to a document upon saving the document such that only the output of the XSLT transformation is saved (col 7, ln 53-41).

As to claim 15, Chau teaches whereby passing an object property to the software application includes passing an object property for controlling whether an XSLT transformation should automatically be applied to a document before the document is saved, such that only the results of the transformation are saved (col 7, ln 53-61).

As to claim 16, Chau teaches whereby passing an object property to the software application includes passing an object property for controlling whether XML markup applied to a document is displayed to the user along with text being edited by the user in a document (col 6, ln 65-67 to col 7, on 1-10/ col 7, ln 52-62).

As to claim 17, Chau teaches passing an object property to the software application includes passing an object property for controlling an appearance of an XML element name as a visible placeholder in a document where no data has been entered for the XML element (col 8, ln 25-30).

As to claim 18, Chau teaches whereby passing an object property to the software application includes passing a method property for inserting XML markup into a document at a specified location, whereby a text string associated with the XML markup to be inserted and whereby any XSLT transformations to be applied to the inserted markup are passed as parameters to the method property (col 7, ln 52- 62).

As to claim 19, Chau teaches passing an object property to the software application includes passing a method object to the software application for creating an XML nodes collection object and for adding to the XNIL nodes collection object any new XML node objects, whereby a name for a new XML node object, a uniform resource identifier identifying a Namespace associated with a new XML node object, and a range pointer to a location in a document where the new XML node object is to be applied are passed to the software application as parameters of the method object(col 8, ln 25-30).

As to claim 20, Chau teaches passing an object property to the software application includes passing a method object for accessing individual XML node objects contained in the collection of XML node objects, whereby an identification representing the position of a requested XML node object in a Namespace library is passed as a parameter of the method object, col 58, ln 55-65).

As to claim 21, Chau teaches passing an object property to the software application includes passing an object property for returning a base name of a specified XML elements (col 7, ln 52-63).

As to claim 22, Chau teaches whereby passing an object property to the software application includes passing an object property for pointing to an XML nodes collection that consists of all XML elements that are child elements for a specified element (col 58, ln 55-56).

As to claim 23, Nussbaum teaches whereby passing an object property to the software application includes passing a method property to the software application for copying a specified XML element and all data associated with the XML element for pasting to a separate location within a document (col 3, ln 50-55).

As to claim 24, Chau teaches passing an object property to the software application includes passing an object property for copying an XML element and all data associated with the XML element and for removing the copied XML element and the copied data associated with the XML element from a document from which the XML element and the data associated with the XML element are copied (col 7, ln 10-15).

As to claim 25, Chau teaches passing an object property to the software application includes passing a method property for removing a specified XML element from a document without affecting data associated with the XML element (col 9, ln 4-8/ col 21, ln 60-64).

As to claim 26, Chau teaches passing an object property to the software application includes passing an object property for pointing to a first XML element that is a child element of a specified XML element (col 30, ln 57-65).

As to claim 27, Chau teaches passing an object property to the software application includes passing an object property for indicating whether an XML element has any child elements that contain no associated data (col 30, ln 60-67).

As to claim 28, Chau teaches passing an object property to the software application includes passing an object property for pointing to a last XNIL element that is a child element of a specified XML element (col 30, ln 60-67).

As to claim 29, Chau teaches passing an object property to the software application includes passing an object property for indicating whether an XML element is at an inline level, a paragraph level, a table cell level, a table row level, a table level, or other levels supported by the software application (col 17, ln 26-40).

As to claim 30, Chau teaches passing an object property to the software application includes passing an object property for returning the uniform resource identifier of an XML schema file associated with a specified XML element (col 8, ln 25-31).

As to claim 31, Chau teaches passing an object property for returning a pointer to an XML node that represents a next XML element immediately following a specified XML element (col 30, ln 55-65).

As to claim 32, Chau teaches passing an object property for indicating a type of an XNIL element, whereby the type of the XML element may include an XML element or an attribute of an XML element (col 2, ln 21028/ col 7, ln 35-45).

As to claim 33, Chau teaches passing an object property to the Software application includes passing an object property for returning a pointer to a document containing a specified XML element (col 30, ln 55-65).

As to claim 34, Chau teaches the method of claim 5, whereby passing an object property to the software application includes passing an object property for returning a pointer to an XML element that is a parent XML element of a specified XML element (col 8, ln 55-65).

As to claim 35, Chau teaches passing an object property to the software application includes passing an object property for controlling whether a placeholder text is displayed in place of XML elements applied to an XML element applied to a document when the XML element contains no associated (col 6, ln 65-67 to col 7, on 1-10/ col 7, ln 52-62).

As to claim 36, Chau teaches passing an object property for returning a pointer to an XML element that is a previous XML element before a specified XML element (col 8, ln 55-65).

As to claim 37, Chau teaches passing an object property to the software application includes passing an object property for returning a pointer to a range in a document contained by a specified XML element (col 8, ln 55-65).

As to claim 38, it is an apparatus claim of claims 25, 26; therefore, they are rejected for the same reasons as claims 25, 26 above.

As to claim 39, Chau teaches to the software application includes passing a method property for finding all XML elements in a document that match a specified XPath query whereby an XML elements collection consisting of all XML elements matching the specified XPath query are returned (col 8, ln 57-62).

As to claim 40, Chau teaches passing an object property to the software application includes passing an object property for returning as plain text any data entered into a document associated with a specified XML element (col 30, ln 55-65).

As to claim 41, Chau teaches passing an object property for returning an XML markup representation of a specified XML element and all data associated with the specified element (col 51, ln 1-7/ col 21, ln 32-37/ col 23, ln 40-45/ col 25, ln 17-23/ col 25, ln 50-55).

Conclusion


Any inquiry concerning this communication or earlier communications from the examiner should be directed to LeChi Truong whose telephone number is (571) 272 3767. The examiner can normally be reached on 8 - 5.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Thomson, William can be reached on (571) 272 3718. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIP. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIP system, contact the Electronic Business Center (EBC) at 866-217-9197(toll-free).

LeChi Truong

June 8, 2007


WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER